

This subroutine has the interface for calling the function *lmder* from MINPACK. *lmder* is a modified levenberg-marquardt algorithm with analytically calculated jacobian.

[called by: [solvers](#).]

[calls: [packdof](#).]

General

Levenberg-Marquardt algorithm is one of the most famous minimization algorithms. There is a brief introduction on Wikipedia. In FOCUS, we use the subroutine *lmder* from MINPACK.

Documentation

The cost functions (targets) are stored in **LM_fvec(1:LM_mfvec)**, and the jacobian is stored in **LM_fjac(1:LM_mfvec, 1:Ndof)**. The number of targets (**LM_mfvec**) must be not be smaller than the number of parameters (**Ndof**).

The targets are consistant of different terms. The following table lists the details. Here are the comments from *lmder*. The four

Table 1: **LM_fvec** components

cost functions	physiccal meaning	switch	length
bnormal	$\mathbf{B} \cdot \mathbf{n}$ on each surface element	weight_bnorm > 0	Nteta*Nzeta
bmnharm	$wBmn_i(Bmn_i - Bmn_i^o)$ for each reasonant harmonics	weight_bharm > 0	2*NBmn in <i>target.harmonics</i>
torflux	$\Psi_i - \Psi_o$ at each toroidal cross-sections	weight_tflux > 0	Nzeta
length	length penalty of each coil, $L_i - L_o$ or $\exp(L_i)/\exp L_o$	weight_ttlen > 0	Ncoils - Nfixgeo
surfsep	potential energy between each coil and a control surface	weight_cssep > 0	Ncoils - Nfixgeo

input parameters for L-M optimizer are **LM_maxiter**, **LM_xtol**, **LM_ftol** and **LM_factor**. Please look at [initial](#) for more details.

```

c      the subroutine statement is
c
c      subroutine lmder(fcn,m,n,x,fvec,fjac,ldfjac,ftol,xtol,gtol,
c                      maxfev,diag,mode,factor,nprint,info,nfev,
c                      njev,ipvt,qtf,wa1,wa2,wa3,wa4)
c
c      where
c
c      fcn is the name of the user-supplied subroutine which
c      calculates the functions and the jacobian. fcn must
c      be declared in an external statement in the user
c      calling program, and should be written as follows.
c
c      subroutine fcn(m,n,x,fvec,fjac,ldfjac,iflag)
c      integer m,n,ldfjac,iflag
c      double precision x(n),fvec(m),fjac(ldfjac,n)
c
c      if iflag = 1 calculate the functions at x and
c      return this vector in fvec. do not alter fjac.
c      if iflag = 2 calculate the jacobian at x and
c      return this matrix in fjac. do not alter fvec.
c
c      return
c      end
c
c      the value of iflag should not be changed by fcn unless
c      the user wants to terminate execution of lmder.
c      in this case set iflag to a negative integer.
c
c      m is a positive integer input variable set to the number
c      of functions.
c
c      n is a positive integer input variable set to the number
c      of variables. n must not exceed m.
c
c      x is an array of length n. on input x must contain
c      an initial estimate of the solution vector. on output x
c      contains the final estimate of the solution vector.
c

```

fvec is an output array of length m which **contains**
the functions evaluated at the output x.

fjac is an output m by n array. the upper n by n submatrix
of fjac **contains** an upper triangular matrix r with
diagonal elements of nonincreasing magnitude such that

$$p \begin{pmatrix} t & & \\ & t & \\ & & t \end{pmatrix} (jac * jac) * p = r * r,$$

where p is a permutation matrix and jac is the final
calculated jacobian. column j of p is column ipvt(j)
(see below) of the identity matrix. the lower trapezoidal
part of fjac **contains** information generated during
the computation of r.

ldfjac is a positive **integer** input variable not less than m
which specifies the leading **dimension** of the array fjac.

ftol is a nonnegative input variable. termination
occurs when both the actual and predicted relative
reductions **in** the sum of squares are at most ftol.
therefore, ftol measures the relative error desired
in the sum of squares.

xtol is a nonnegative input variable. termination
occurs when the relative error between two consecutive
iterates is at most xtol. therefore, xtol measures the
relative error desired **in** the approximate solution.

gtol is a nonnegative input variable. termination
occurs when the cosine of the angle between fvec and
any column of the jacobian is at most gtol **in** absolute
value. therefore, gtol measures the orthogonality
desired between the **function** vector and the columns
of the jacobian.

maxfev is a positive **integer** input variable. termination
occurs when the **number** of calls to fcn with iflag = 1
has reached maxfev.

diag is an array of length n. **if** mode = 1 (see
below), diag is internally set. **if** mode = 2, diag
must contain positive entries that serve as
multiplicative scale factors for the variables.

mode is an **integer** input variable. **if** mode = 1, the
variables will be scaled internally. **if** mode = 2,
the scaling is specified by the input diag. other
values of mode are equivalent to mode = 1.

factor is a positive input variable used **in** determining the
initial step bound. this bound is set to the product of
factor and the euclidean norm of diag*x **if** nonzero, or **else**
to factor itself. **in** most cases factor should lie **in** the
interval (.1,100.).100. is a generally recommended value.

nprint is an **integer** input variable that enables controlled
printing of iterates **if** it is positive. **in** this **case**,
fcn is called with iflag = 0 at the beginning of the first
iteration and every nprint iterations thereafter and
immediately prior to **return**, with x, fvec, and fjac
available for printing. fvec and fjac should not be
altered. **if** nprint is not positive, no special calls
of fcn with iflag = 0 are made.

info is an **integer** output variable. **if** the user has
terminated execution, info is set to the (negative)
value of iflag. see description of fcn. otherwise,
info is set as follows.

```

c      info = 0  improper input parameters.
c
c      info = 1  both actual and predicted relative reductions
c                in the sum of squares are at most ftol.
c
c      info = 2  relative error between two consecutive iterates
c                is at most xtol.
c
c      info = 3  conditions for info = 1 and info = 2 both hold.
c
c      info = 4  the cosine of the angle between fvec and any
c                column of the jacobian is at most gtol in
c                absolute value.
c
c      info = 5  number of calls to fcn with iflag = 1 has
c                reached maxfev.
c
c      info = 6  ftol is too small. no further reduction in
c                the sum of squares is possible.
c
c      info = 7  xtol is too small. no further improvement in
c                the approximate solution x is possible.
c
c      info = 8  gtol is too small. fvec is orthogonal to the
c                columns of the jacobian to machine precision.
c
c      nfev is an integer output variable set to the number of
c                calls to fcn with iflag = 1.
c
c      njev is an integer output variable set to the number of
c                calls to fcn with iflag = 2.
c
c      ipvt is an integer output array of length n. ipvt
c                defines a permutation matrix p such that jac*p = q*r,
c                where jac is the final calculated jacobian, q is
c                orthogonal (not stored), and r is upper triangular
c                with diagonal elements of nonincreasing magnitude.
c                column j of p is column ipvt(j) of the identity matrix.
c
c      qtf is an output array of length n which contains
c                the first n elements of the vector (q transpose)*fvec.
c
c      wa1, wa2, and wa3 are work arrays of length n.
c
c      wa4 is a work array of length m.
c
c      subprograms called
c
c      user-supplied ..... fcn
c
c      minpack-supplied ... dppar,enorm,lmpar,qrfac
c
c      fortran-supplied ... dabs,dmax1,dmin1,dsqrt,mod
c
c      argonne national laboratory. minpack project. march 1980.
c      burton s. garbow, kenneth e. hillstom, jorge j. more

```